

What is claimed is:

1. A method used in computer-aided circuit design for splitting the data flow graph of a finite-precision arithmetic circuit into a plurality of lossless subgraphs wherein each /  
lossless subgraph represents an infinite precision arithmetic circuit, the data flow  
5 graph comprising a plurality of edges wherein each edge has a finite bit-width, an edge being further connected to a source port and one or more sink ports, the source port and the sink ports linked to an edge further connecting the edge to a plurality of arithmetic operators, each port further having a required precision and an information content associated with it, the required precision of a port being the  
10 maximum number of bits of the port observed at any of the outputs of the port, the information content of a port being the minimum number of bits required to represent the output of an arithmetic operation, the method comprising the steps of:
  - a. comparing the information content of the sink port, the required precision of the source ports, of each edge, to the bit width of the edge;
  - 15 b. determining the edges having information loss, an edge has an information loss if the information content of the corresponding source port and the required precision of any of the corresponding sink ports of the edge are greater than the bit width of the edge; and
  - c. dividing the data flow graph at edges having information loss, the data flow  
20 graphs being split into lossless subgraphs wherein each lossless subgraph representing an infinite-precision arithmetic circuit.
2. The method as recited in claim 1 further comprising the steps of:

a. computing the required precision of the source port and the sink ports  
corresponding to each edge wherein the required precision is computed in the  
reverse topological order of the circuit; and

b. determining the information content of the source port and the sink ports  
corresponding to each edge wherein the information content is computed in the  
topological order of the circuit.

3. The method as recited in claim 2 wherein the step of computing the required  
precision comprises the steps of:

a. determining the required precision of the sink port of the output signal as the bit  
width of the signal itself;

b. calculating the minimum of the required precision of the sink port, and the bit  
width of the corresponding edge for all sink ports of an edge;

c. computing the required precision of the source port as the maximum of the set of  
minimum values obtained, the minimum values obtained by comparing required  
precision of the sink port and the bit width of the corresponding edge for all sink  
ports of an edge; and

d. determining the required precision at each sink port linked to an arithmetic  
operator, wherein the required precision is determined, using the required  
precision at the source port, as the maximum number of bits observable at the  
output edge;

4. The method as recited in claim 2 wherein the step of determining the information content comprises the steps of:

a. computing the information content at the source port of the input signal, as the bit-width of the signal;

5 b. calculating the information content of each sink port as the minimum of the information content of the corresponding source port, and the bit width of the corresponding edge; and

c. computing the information content at a source port linked to an arithmetic operator, using the information content at the sink ports linked to the arithmetic operation, as the minimum of the bit width required to represent the output of the corresponding operator.

10 5. A method used in computer-aided circuit design for comparing data flow graphs for equivalence, each data flow graph representing a finite-precision arithmetic circuit, the data flow graph comprising a plurality of edges wherein each edge has a finite bit width, an edge being further connected to a source port and one or more sink ports, the source port and the sink ports linked to an edge further connecting the edge to a plurality of arithmetic operators, each port further having a required precision and an information content associated with it, the required precision of a port being the maximum bits of the port observed at any of the output of the port, the information content of a port being the minimum bits required to represent the output of an arithmetic operation, the method comprising the steps of:

- a. determining the edges having information loss wherein an edge has an information loss if the information content of the corresponding source port, and the required precision of any of the corresponding sink ports are greater than the bit-width of the edge;
- 5      b. canonizing the data flow graph, the data flow graph being canonized by pushing the arithmetic operations that generate information loss, to the end of the circuit;
- c. dividing the data flow graph at edges having information loss, the data flow graphs being split into lossless subgraphs, each lossless subgraph representing an infinite-precision arithmetic circuit;
- 10     d. refining the information content of the ports in the data flow graphs, the information content being refined by applying the Huffman Principle and the principles of associativity of addition and multiplication, and the distributivity of multiplication over addition;
- 15     e. verifying the presence of information loss, the information loss being verified at edges corresponding to the ports for which the refinement of information content is performed;
- f. leveling each data flow graph, the level being the order of existence of lossless subgraphs in the data flow graph whereby each data flow graph is leveled for comparison of subgraphs at the same level;

g. comparing the lossless subgraphs with lowest level number for equality wherein the lossless subgraphs being compared only if the input signal to the data flow graphs have the same bit-width;

5 h. comparing the bit-width of the output edges of the lossless subgraphs, the output edges having information loss, wherein the bit-width of the output edges being compared if the preceding lossless subgraphs are compared equal;

i. comparing the lossless subgraphs with next higher level number for equivalence wherein the comparison of next set of lossless subgraphs being performed if the preceding lossless subgraphs are equal and the bit width of the preceding edges  
10 are same; and

repeating the steps h and i till the output of the circuit is reached.

6. The method as recited in claim 5 further comprises the steps of:

a. computing the required precision of the source port and the sink ports corresponding to each edge wherein the required precision is computed in the  
15 reverse topological order of the circuit; and

b. determining the information content of the source port and the sink ports corresponding to each edge wherein the information content being computed in the topological order of the circuit.

7. The method as recited in claim 6 wherein the step of computing the required  
20 precision comprises the steps of:

a. determining the required precision of the sink port of the output signal as the bit width of the signal itself;

b. determining the required precision at each sink port linked to an arithmetic operator, wherein the required precision is determined as the maximum number of bits observable at the output edge;

c. calculating the minimum of the required precision of the sink port, and the bit width of the corresponding edge for all sets of sink ports of an edge; and

d. computing the required precision of the source port as the maximum of the set of minimum values obtained, the minimum values being obtained by comparing the required precision of the sink port and the bit width of the corresponding edge for all sets of sink ports of an edge.

8. The method as recited in claim 6 wherein the step of determining the information content comprises the steps of:

a. computing the information content at the source port of the input signal as the bit width of the signal;

b. calculating the information content of each sink port as the minimum of the information content of the corresponding source port, and the bit width of the corresponding edge; and

c. computing the information content at a source port linked to an arithmetic operator using the information content at the sink ports linked to the arithmetic

operation, as the minimum bit width required to represent the output of the corresponding operator.

9. The method as recited in claim 5 wherein the step of leveling each data flow graph comprises allocating identifiers to lossless subgraphs wherein the identifiers being  
5 allocated to lossless subgraphs in topological order.

10. The method as recited in claim 5 wherein the step of comparing the lossless subgraphs for equality comprises the steps of:

- a. generating expressions for the data flow graphs; and
- b. checking the equality of expressions, the expressions being checked by theorem  
10 proving technique.

11. The method as recited in claim 10 wherein the step of checking the equality of expressions comprises the step of testing the equality using Cooperating Validity Checker Lite (CVC Lite) program.

12. A system used in computer-aided circuit design for testing equivalence of a plurality  
15 of data flow graphs, each data flow graph representing a finite-precision arithmetic circuit, the data flow graph comprising a plurality of edges wherein each edge has a finite bit-width, an edge being connected to an arithmetic operator through a source and one or more sink port wherein each port having a required precision and an information content associated with it, the required precision of a port being the  
20 maximum bits of the port observed at any of its output, the information content of a

port being the minimum bits required to represent the output of an arithmetic operation, the system comprising:

a. means for computing required precision and information content;

b. means for canonizing a data flow graph;

5 c. means for generating data flow graphs and the corresponding lossless subgraphs;

d. means for leveling the lossless subgraphs;

e. means for generating expressions;

f. means for logical comparison of expressions; and

10 g. means for comparing bit-width of the edges having information loss.

13. The system as described in claim 12 further comprises memory means for storing the bit widths of edges, the required precision and the information loss at the source port, and the sink ports of each edge, and word level expression.

14. A computer program product, the computer program product comprising a computer  
15 usable medium having a computer readable program code embodied therein for  
splitting the data flow graph of a finite-precision arithmetic circuit into a plurality of  
lossless subgraphs, wherein each lossless subgraph represents an infinite precision  
arithmetic circuit, the data flow graph comprising a plurality of edges wherein each  
edge has a finite bit-width, an edge being further connected to a source port and one  
20 or more sink ports, the source port and the sink ports linked to an edge further



connecting the edge to a plurality of arithmetic operators, each port further having a required precision and an information content associated with it, the required precision of a port being the maximum number of bits of the port observed at any of the outputs of the port, the information content of a port being the minimum number of bits required to represent the output of an arithmetic operation, the computer program code performing steps of:

- a. comparing the information content of the sink port, the required precision of the source ports, of each edge, to the bit width of the edge;
- b. determining the edges having information loss, an edge has an information loss if the information content of the corresponding source port and the required precision of any of the corresponding sink ports of the edge are greater than the bit width of the edge; and
- c. dividing the data flow graph at edges having information loss, the data flow graphs being split into lossless subgraphs wherein each lossless subgraph representing an infinite-precision arithmetic circuit.

15. The computer program product described in claim 14, wherein the computer program code further performs the steps of:

- a. computing the required precision of the sink port and the source ports corresponding to each edge wherein the required precision is computed in the reverse topological order of the circuit; and

- b. determining the information content of the source port and the sink ports corresponding to each edge wherein the information content is computed in the topological order of the circuit.

16. A computer program product, the computer program product comprising a computer

5       usable medium having a computer readable program code embodied therein for  
comparing data flow graphs for equivalence, each data flow graph representing a  
finite-precision arithmetic circuit, the data flow graph comprising a plurality of edges  
wherein each edge has a finite bit width, an edge being further connected to a  
source port and one or more sink ports, the source port and the sink ports linked to  
10       an edge further connecting the edge to a plurality of arithmetic operators, each port  
further having a required precision and an information content associated with it, the  
required precision of a port being the maximum bits of the port observed at any of  
the output of the port, the information content of a port being the minimum bits  
required to represent the output of an arithmetic operation, the computer program  
15       code performing steps of:

- a. determining the edges having information loss wherein an edge has an information loss if the information content of the corresponding source port, and the required precision any of the corresponding sink ports are greater than the bit-width of the edge;
- 20       b. canonizing the data flow graph, the data flow graph being canonized by pushing the arithmetic operations that generate information loss, to the end of the circuit;

- c. dividing the data flow graph at edges having information loss, the data flow graphs being split into lossless subgraphs, each lossless subgraph representing an infinite-precision arithmetic circuit;
- d. refining the information content of the ports in the data flow graphs, the  
5 information content being refined by applying the Huffman Principle and the principles of associativity of addition and multiplication, and the distributivity of multiplication over addition;
- e. verifying the presence of information loss, the information loss being verified at  
10 edges corresponding to the ports for which the refinement of information content is performed;
- f. leveling each data flow graph, the level being the order of existence of lossless subgraphs in the data flow graph whereby each data flow graph is leveled for comparison of subgraphs at the same level;
- g. comparing the lossless subgraphs with lowest level number for equality wherein  
15 the lossless subgraphs being compared only if the input signal to the data flow graphs have the same bit-width;
- h. comparing the bit-width of the output edges of the lossless subgraphs, the output edges having information loss, wherein the bit-width of the output edges being compared if the preceding lossless subgraphs are compared equal; and
- 20 i. comparing the lossless subgraphs with next higher level number for equivalence wherein the comparison of next set of lossless subgraphs being performed if the

preceding lossless subgraphs are equal and the bit width of the preceding edges are same.

17. The computer program product described in claim 16, wherein the computer program code further performs the steps of:

- 5       a. computing the required precision of the source port and the sink ports  
corresponding to each edge wherein the required precision is computed in the  
reverse topological order of the circuit; and
- b. determining the information content of the source port and the sink ports  
corresponding to each edge wherein the information content being computed in  
10      the topological order of the circuit.